ROBUST PANORAMA FROM MPEG VIDEO

Yongmin Li, Li-Qun Xu, Geoff Morrison, Charles Nightingale and Jason Morphett

BTexact Technologies, pp2 Ross Building, Adastral Park, Ipswich IP5 3RE, UK
Email: Yongmin.Li@bt.com

ABSTRACT

A novel approach to image mosaicking from MPEG video is presented in this paper. The motion vectors in both P- and B-frames are used for global motion estimation. The bi-directional information in B-frames provides multiple routes to warp a frame to its previous anchor frame. A Least Median of Squares based algorithm is adopted for robust motion estimation. In the case of a large proportion of outliers, we detect possible algorithm failure and perform re-estimation along a different route. Based on the motion parameters between consecutive frames, the static background panorama and dynamic foreground panorama are constructed from warped images over a whole video sequence.

1. INTRODUCTION

Panoramic scene reconstruction [1, 2, 3] has been an interesting research topic for several decades. By warping a sequence of images onto a single reference mosaic image, we not only obtain an overview of the content across the whole sequence but also reduce the spatio-temporal redundancy in the original sequence of images. Image registration, i.e. establishing the correspondence between images, is one of the most computationally intensive stages for image panorama. Fortunately, MPEG video, which has been widely available in many applications such as teleconferencing, visual surveillance, video-on-demand and VCDs/DVDs, has pre-encoded macroblock based motion vectors that are potentially useful for image registration.

There has been considerable effort over the past few years in constructing panorama from an MPEG video [4, 5, 6, 7, 8]. However, the motion information encoded in an MPEG video, especially the bi-directional information in B-frames, has not been fully utilised in the previous studies. Another limitation is the methods adopted are mostly based on least squares minimisation which is not robust to "outlying" MPEG motion vectors. Although some researchers have proposed some methods accordingly, e.g. texture filtering [6] and M-estimation [8], the problem is still largely unsolved.

In this paper, we present a novel approach to image panorama from MPEG video. The motion vectors in both P- and B-frames of an MPEG video are used for motion estimation. The bi-directional information in B-frames provides multiple routes to warp a frame to its previous anchor frame. As the MPEG motion vectors are usually noisy and contain many outliers, a robust Least Median of Squares (LMedS) algorithm is adopted. In case of large proportion of outliers, we detect possible algorithm failure and perform re-estimation along a different route. Finally, the static background panorama and dynamic foreground panorama are constructed from warped images over a whole video sequence.

The rest of the paper is arranged as follows: Section 2 discusses the basic methods of estimating global motion from MPEG motion vectors and maintaining the continuity of motion estimation. A robust LMedS based algorithm is presented in Section 3. Section 4 discusses static background panorama and dynamic foreground panorama construction. Section 5 concludes the paper.

2. GLOBAL MOTION ESTIMATION FROM MPEG MOTION VECTORS

MPEG (MPEG1, MPEG2 and MPEG4) is a family of motion prediction based compression standards. Three types of pictures, I-, P- and B-pictures are defined by MPEG, where an I-picture is coded entirely in intra mode, a P-picture is coded using motion prediction from the previous I- or P-picture (forward prediction), and a B-picture is coded using forward prediction or backward prediction or both. I- and P-pictures are often referred to as anchor frames as they may be reference frames of other B- and P-pictures.

Although these MPEG motion vectors are encoded for the purpose of video compression and may not be the real motion vectors, we would argue that, given a MPEG video with reasonable image and compression quality, most MPEG motion vectors are likely to reflect the underlying global motion (or camera motion) in a video. Therefore it is possible to estimate the global motion from MPEG motion vectors.

2.1. Motion Estimation from MPEG Motion Vectors

We assume the global motion can be modelled as a 6-parameter affine transformation given by

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \tag{1}$$

where $(x, y)^T$ and $(x', y')^T$ are the 2D positions before and after transformation, and $a_1, a_2, a_3, a_4, b_1, b_2$ are parameters of the affine transformation. When more than 3 motion vectors between two frames are available, this transformation can be estimated using a least squares method. Denote the parameters of the affine transformation as a column vector

$$\mathbf{a} = (a_1, a_2, b_1, a_3, a_4, b_2)^T \tag{2}$$

For the training vectors pair $(x_i, y_i)^T$ and $(x'_i, y'_i)^T$, we define

$$X_i = \begin{bmatrix} x_i & y_i & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_i & y_i & 1 \end{bmatrix} \tag{3}$$

$$Y_i = \begin{bmatrix} x'_i \\ y'_i \end{bmatrix} \tag{4}$$

Then the least squares solution to this problem is given by

$$\mathbf{a} = \left(\sum_i X_i^T X_i \right)^{-1} \left(\sum_i X_i^T Y_i \right) \tag{5}$$

When the affine transformations between all pairs of consecutive frames are available, the whole video sequence can be warped to a reference frame, e.g. the first frame of the sequence. A 2D position vector in the first frame, $\mathbf{x}_0 = (x_0, y_0)^T$ is transformed to

$$\mathbf{x}_n = f_n(\mathbf{x}_{n-1}) \tag{6}$$

in the n-th frame, where f_n is the affine transformation between the n-th and $(n-1)$-th frames given by (1). Thus the pixel value of $\mathbf{x}_0$ in the first frame is taken as that of $\mathbf{x}_n$ in the n-th frame.

We have also experimented with a more complicated projective transformation with 8 parameters. However, the results were not as good as those of the simple affine transformation (for example, larger distortion, image features like lines not aligned well, etc.), which indicate that complicated models may not be necessary for the "noisy" MPEG motion vectors.

2.2. Continuity of Motion Estimation

To achieve an image panorama over a video sequence, we need a continuous motion from the beginning to the end of the sequence. It is not a problem for the conventional panorama methods based on motion estimation from raw images. However, for the MPEG motion vectors, we may have the following problems: (1) The motion would be no longer continuous at an I-frame if its immediate preceding frame is not a B-frame; and (2) intrinsically the MPEG motion vectors do not need to reflect the real motion as long as a decoder is able to re-construct the image in an acceptable quality. Thus these motion vectors may appear too noisy for motion estimation.

To address these problems, we adopt several strategies which proved to be useful for panorama from MPEG video.

1. Exclude the motion vectors along the boundaries of an image which tend to be errant.
2. Exclude the zero motion vectors which usually do not specify a static macroblock in MPEG.
3. Owing to the bi-directional motion vectors in B-pictures, there may be multiple routes to warp a frame to its preceding anchor frame (I- or P-frame) and ultimately to the reference frame. For example, given four consecutive frames $I_1B_2B_3P_4$, there are three different routes from frame P_4 to I_1: $P_4 - I_1$, $P_4 - B_3 - I_1$ and $P_4 - B_2 - I_1$. We can then perform motion estimation along all the routes and select the best as the final result.
4. If none of the possible routes for a given frame provides a satisfactory estimation, then
   (a) this frame is removed from mosaic construction if it does not affect the continuity of motion estimation, e.g. frame B_2 in the previous example;
   (b) the transformation between this frame to its preceding frame is interpolated from the transformations of its neighbouring frames.
5. Use robust algorithms to estimate the global motion, which will be discussed in Section 3.

3. ROBUST MOTION ESTIMATION

Robust techniques have been widely used in computer vision problems. A robust method should provide reliability in the presence of various types of noise and can tolerate a certain portion of outliers [9, 10]. Interested readers may refer to [11] for a recent review on robust methods in computer vision.

3.1. LMedS Algorithm

To demonstrate the necessity of using robust methods for global motion estimation from MPEG video, let us begin with a set of typical motion vectors from a P- and B-frame in a football video as shown in Figure 1(a), (b) and (c). As this image is taken from a long distance and contains a dominant static ground-plane, most motion vectors reflect the global camera motion. However, a few motion vectors look different from the majority owing to the foreground object motion or MPEG encoding efficiency as discussed previously. These extraordinary motion vectors should be treated as outliers for global motion estimation. It is important to note that the outlier vectors are more likely to have large magnitudes, therefore they may easily skew the solution from the desired one if they are dealt with inappropriately.

We adopt the robust Least Median of Squares (LMedS) method [12] for global motion estimation. The outline of the method is as follows.

1. Randomly select N sets of data from all available training examples to fit the model, resulting in N candidate solutions;
2. Rather than using as much of the data as possible, each randomly selected data set only contains s data points, the minimum number to sufficiently solve the problem;
3. The optimal solution is chosen as the one with least median of squared error.

Given an expected proportion of outliers in the data (ϵ, say) then we need to choose N sufficiently large to give a good probability (p, say) of having at least one set which does not contain an outlier. By simple probability it is easy to show that N can be calculated from the formula:

$$N = log(1-p)/log(1-(1-\epsilon)^s) \tag{7}$$

where p is the probability of at least one of N random samples is free from outliers, ϵ if the expected proportion of outliers in the training data, i.e. the probability of a data point being an outlier, and s is the sample size. For our problem of affine motion estimation, the minimum sample size $s = 3$. Even if we make a very conservative decision by choosing $p = 0.99$ and $\epsilon = 50\%$, then $N = 34$ which is still feasible for a good real-time performance.

3.2. Algorithm Failure Control

The LMedS method is very simple and does not need any prior knowledge of the problem. However, its main shortcoming is that when more than half of the training data are outliers, i.e. $\epsilon > 50\%$, the data point with the median value may be an outlier, therefore the algorithm will fail in this case. However, given a frame with its motion vectors, we do not know if the condition $\epsilon < 50\%$ holds.

For our specific problem, we can use the following methods to solve the above problem:

1. Estimate motion along multiple routes and select the one with smallest error. This would considerably reduce the possibility of algorithm failure.
2. Use a threshold T to determine a possible failure of the LMedS algorithm, i.e. if the median of squares is larger than T, an estimation failure is raised. The strategies described in Section 2.2 (3 and 4) are then adopted to compute an alternative solution, i.e. computing along a different route, dropping the frame, or interpolating from neighbouring frames.

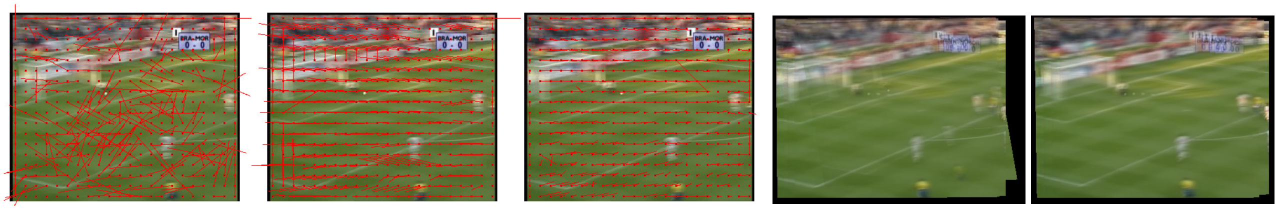

(a) (b) (c) (d) (e)

Fig. 1. Motion estimation along multiple routes. The concerned frames are 144-147 (IBBP). The forward motion vectors of frame 147 to its previous anchor frame 144 (a) are too noisy for a reasonable estimation (d). However, using both the forward (b) and backward (c) motion vectors in frame 146, a satisfactory motion estimation between frame 147 and 144 can still be established (e).

It is important to note that determining the value of T is by no means a tricky practice, at least for the data we used in this work. The unreliable estimations can be easily distinguished from the good ones. For example, a threshold $T = 18$, which is not a magic number but has an analytical meaning that less than 3 pixel displacement in both horizontal and vertical direction is acceptable $(3^2 + 3^2)$, proved to work fairly well.

3.3. Algorithm Description

Combining the idea of LMedS and the strategies of determining possible failures of the LMedS algorithm (Section 3.1) and maintaining the continuity of motion estimation (Section 2.2), we give, as in Table 1, the formal description of our algorithm for robust global motion estimation from MPEG video.

1. Decode the motion vectors from the input MPEG video;
2. Remove the zero motion vectors and those on the boundaries of an image;
3. Randomly select N sets of motion vectors from the remaining data with sample size 3, where N is computed with (7);
4. Compute the affine transformation for each of the N sample sets (5);
5. Compute the median of squared error for each of the N transformation, and select the one with the smallest value Med;
6. If $Med < T$, return its corresponding parameters as the optimal solution;
7. Otherwise, repeat the above steps along a different route as described in Section 2.2, and return the optimal solution if no failure raised;
8. If algorithm fails along all possible routes, drop the frame or interpolate from the affine transformations of neighbouring frames.

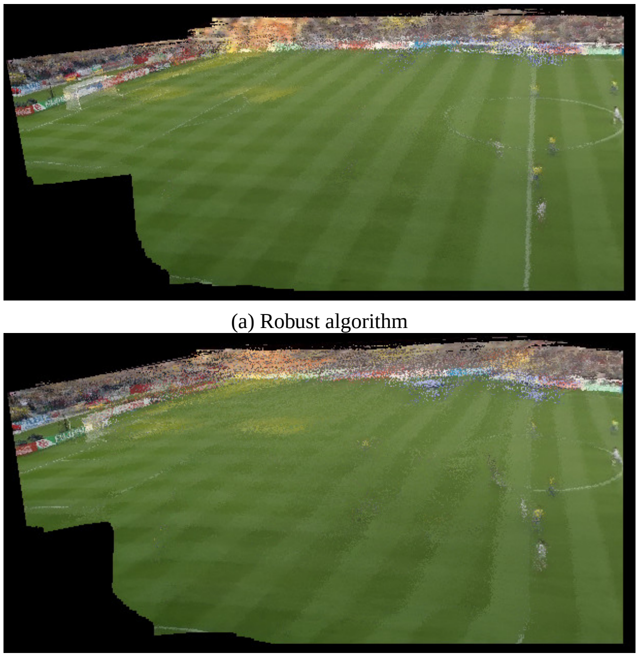
Table 1. The algorithm of robust global motion estimation from MPEG video.

Note that, although it may be better to perform motion estimation along all possible routes and select the best result, it is computationally more efficient if we simply select the first satisfactory result with $Med < T$ as in Step 6. Also note the order of routes to compute: For an I-frame, we first select its immediate preceding B-frame, and decode the backward motion vectors of this B-frame to estimate the global motion. If a failure raised, we then select the second immediate preceding B-frame, and so on. For a P-frame, the order is its preceding anchor frame, first immediate preceding B-frame, second immediate preceding B-frame and so on. A B-frame is usually directly warped to its preceding anchor frame. Here are a few examples:

I-frame	in $\cdots B_3 B_2 B_1 I \cdots$,	order $B_1 B_2 B_3$
P-frame	in $\cdots I B_2 B_1 P \cdots$,	order $I B_1 B_2$

3.4. Results

We have applied the above algorithm to different MPEG videos. Figure 2(a) shows the panoramic image constructed from a football video clip using our robust algorithm. To compare its performance to other non-robust methods, a panorama using the standard least squares method is also shown in (b). It is noted that there are apparent distortions in the image constructed using least squares method, while that using robust method looks much better.


(a) Robust algorithm

(b) Standard least squares method

Fig. 2. Panorama constructed from a football video clip. Distortions can be observed from the panorama using the standard least squares method (b), while the robust method achieves a more accurate result (a).

Figure 1 demonstrates the situation of motion estimation along multiple routes which we have been discussed previously. The concerned frames are 144-147 (IBBP). Owing to fast motion, the forward motion vectors of frame 147 (P-frame) to the previous anchor frame (frame 144, I-frame) contain too many outliers for a reasonable estimation as shown in Figure 1(a). More precisely, the least median of squared error $Med = 791.8$, resulting in a failure raised with the algorithm set up as above. This means we cannot warp the current frame to its previous anchor frame directly. Fortunately both the backward and forward motion vectors in frame 146, its immediate preceding B-frame as shown in Figure 1(b) and (c), are sufficiently clean. Therefore we can warp the current frame to its previous anchor frame through two consecutive affine transformations estimated from the forward and backward motion vectors of that B-frame respectively (with $Med = 3.4$ and $Med = 1.4$ respectively). The panoramic images by warping the 4 neighbouring frames to frame 144 using the direct route and indirect route are compared in Figure 1(d) and (e). Pixels in the panoramic images are computed as average values. It is clear that using algorithm failure control and estimating the global motion along an alternative route give a more accurate result.

4. BACKGROUND AND FOREGROUND PANORAMAS

The content contained in a video sequence includes the static (background) and dynamic (foreground) information. The conventional method of computing foreground and background panoramas is to take the mismatched pixels as foreground and matched pixels background. Here we use a simpler and more efficient method: the background panorama is constructed from the medians of accumulated pixels from all frames of a video sequence, while the foreground panorama is comprised of the most different pixels from the mean.

Suppose there are M accumulated values for a pixel position in the panoramic image. The mean RGB values are expressed as

$$\bar{r} = \frac{1}{M}\sum_{i=1}^{M} r_i, \quad \bar{g} = \frac{1}{M}\sum_{i=1}^{M} g_i, \quad \bar{b} = \frac{1}{M}\sum_{i=1}^{M} b_i \tag{8}$$

Compute the L1 distance, which is usually more robust than the L2 distance [9], between each accumulated pixel value (r_i, g_i, b_i) and the mean value $(\bar{r}, \bar{g}, \bar{b})$

$$d_i = |r_i - \bar{r}| + |g_i - \bar{g}| + |b_i - \bar{b}| \tag{9}$$

Then the pixel value with the median of $\{d_i, i = 1, ..., M\}$ is selected for the background panorama, while the one with the largest d_i, i.e. the most different pixel, is selected for the foreground panorama. An example foreground panorama constructed from a football video clip is shown in Figure 3 while its background panorama has been shown in Figure 2(a). Note that the trajectories of both the players and the ball are clearly displayed in the foreground panorama. It is not difficult to understand the whole process of the goal from the single panoramic image.

5. CONCLUSIONS

We have presented an approach to the problem of robust panorama construction from MPEG video. The novel contributions of this work include:

1. Making full use of both the forward and backward motion vectors from P-pictures and B-pictures. As there are bi-directional motion vectors coded in B-frames, motion estimation can be performed along multiple routes for an optimal result.
2. Realising that there likely many outliers existing in the MPEG motion vectors, we adopt a robust LMedS algorithm that theoretically has a high breakdown point of 50%. Algorithm failure control by using multiple routes and thresholding is designed to deal with the case of more than 50% motion vectors being outliers.
3. A simplified method to construct panoramic background and foreground is presented: the median of the accumulated pixels is selected as background pixel and the most different pixel from the mean as foreground.

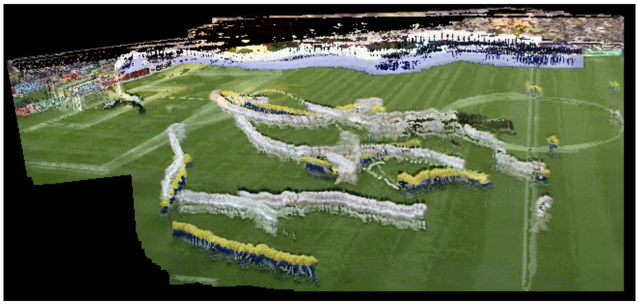


Fig. 3. Foreground panorama constructed from a football video clip. Its background panorama is shown in Figure 2(a).

6. REFERENCES

[1] Harpreet Sawhney, Serge Ayer, and Monika Gorkani, "Model-based 2D&3D dominant motion estimation for mosaicing and video representation," in *IEEE International Conference on Computer Vision*, Cambridge, MA, USA, 1995.

[2] Shmuel Peleg and Joshua Herman, "Panoramic mosaics by manifold projection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1997.

[3] Michal Irani and P. Anandan, "Video indexing based on mosaic representations," *Proceedings of the IEEE*, vol. 86, no. 5, pp. 905–921, 1998.

[4] J. Meng and S. Chang, "CVEPS - a compressed video editing and parsing system," in *ACM Multimedia*, 1996.

[5] Y. Tan, D. Saur, and S. Kulkarni, "Rapid estimation of camera motion from compressed video with application to video annotation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, no. 1, pp. 133–146, 2000.

[6] Maurizio Pilu, "On using raw mpeg motion vectors to determine global camera motion," in *SPIE Electronic Imaging Conference*, San Jose, 1998.

[7] R. Jones, D. DeMenthon, and D. Doermann, "Building mosaics using mpeg motion vectors," in *ACM Multimedia*, 1999.

[8] A. Smolic, M. Hoeynck, and J.-R. Ohm, "Low-complexity global motion estimation from P-frame motion vectors for MPEG-7 application," in *IEEE International Conference on Image Processing*, Vancouver, Canada, September 2000.

[9] Peter J. Huber, *Robust Statistics*, John Wiley & Sons Inc, 1981.

[10] F.R. Hampel, E.M. Ronchetti, P.J. Rousseeuw, and W.A. Stahel, *Robust Statistics*, John Wiley & Sons Inc, 1986.

[11] Peter Meer, Charles V. Stewart, and David E. Tyler, "Robust computer vision: an interdisciplinary challenge," *Computer Vision and Image Understanding*, vol. 78, pp. 1–7, 2000.

[12] P.J.Rousseeuw, "Least median of squares regression," *Journal of The American Statistical Association*, vol. 79, pp. 871–880, 1984.